

Certificate of Electronic Submission  
I hereby certify that this correspondence is being  
submitted electronically to the U.S. Patent  
and Trademark Office on August 24, 2006  
(Date of Submission)

FRANK C. NICHOLAS (33,983)  
Name of applicant, assignee or registered representative

/Frank C. Nicholas/

Signature

August 24, 2006

Date of Signature

PATENT  
Case No. AUS920000800US1  
(9000/10)

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re patent application of:	)	
	)	
STEVEN M. FRENCH, ET AL.	)	Examiner: NGUYEN, T
	)	
Serial No.: 09/731,631	)	
	)	Group Art Unit: 2144
Filed: DECEMBER 7, 2000	)	
	)	
Title: METHOD AND SYSTEM FOR	)	Conf. No. 1071
SELECTING AN OPERATING SYSTEM	)	
AT USER LOGIN ON A TARGET DEVICE	)	

**REVISED APPEAL BRIEF**

Mail Stop Appeal Briefs - Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Dear Sir:

Appellants herewith respectfully present their appeal brief as follows:

**TABLE OF CONTENTS**

1.	Real Party In Interest . . . . .	3
2.	Related Appeals And Interferences . . . . .	4
3.	Status Of Claims . . . . .	5
4.	Status Of Amendments . . . . .	6
5.	Summary Of Claimed Subject Matter. . . . .	7
6.	Grounds Of Rejection To Be Reviewed On Appeal . . . . .	10
7.	Arguments . . . . .	11
8.	Summary . . . . .	17
9.	Claims Appendix . . . . .	18
10.	Evidence Appendix . . . . .	22
11.	Related Proceedings Appendix . . . . .	22

1. REAL PARTY IN INTEREST

The real party in interest remains Assignee INTERNATIONAL BUSINESS  
MACHINES CORPORATION, by virtue of an assignment executed by the inventors.

2. RELATED APPEALS AND INTERFERENCES

Appellants and the undersigned attorneys are not aware of any appeals or any interferences which will directly affect or be directly affected by or having a bearing on the Board's decision in the pending appeal. Appellants do note that two appeals are currently pending before the Board that are related to the instant case, but the patentability of each is independent and the outcome of any of these matters has no bearing on the outcome of the others. *See*, United States Patent Applications 09/731,624 and 09/731,629.

August 24, 2006

Case No.: AUS920000800US1 (9000/10)

Serial No.: 09/731,631

Filed: December 7, 2000

Page 5 of 22

3. STATUS OF CLAIMS

Claims 1-23 stand rejected as unpatentable over Beelitz, et. al., United States Patent No. 6,182,275B1 in view of Barr et. al, United States Patent 6,189,100.

Claims 1-23 are the claims on appeal. *See*, Appendix.

4. STATUS OF AMENDMENTS

All amendments have been entered.

## 5. SUMMARY OF CLAIMED SUBJECT MATTER

In this summary of claimed subject matter, all citations are to the specification of United States Patent Application 09/731,631 filed on December 7, 2000. Further, all citations are illustrative only and support for the cited element may be found elsewhere in the specification. See, e.g. FIGS. 1, 4, 5, and 6.

### **Claim 1**

The invention relates to a method of selecting an operating system at a target device 108, 110, 112 in communication with a server 104, 105. The method includes initiating 402 a network bootstrap program at the target device 108, 110, 112 and sending 406 a bootstrap list command from the target device 108, 110, 112 to the server 104, 105 (p. 9, lines 15-28). The method further includes receiving 516 an operating systems list of at least one operating system prior to executing an operating system at the target device 108, 110, 112, and selecting 518 a target operating system from the operating systems list. In this aspect of the invention, the target device 108, 110, 112 is to be remotely booted by the server 104, 105 (p. 1, lines 12-15).

### **Claim 6**

In one embodiment, the method further includes relocating 524 the network bootstrap program after the target operating system is selected. (p. 11, lines 23-25)

### **Claim 8**

In another embodiment, the target operating system is selected by a user (p.11, lines 13-18) of the target device 108, 110, 112.

### **Claim 9**

In yet another embodiment, the method further includes determining from a user profile, at least one available operating system and including the user-available operating system with the operating systems list (p. 12, lines 8-20).

**Claim 11**

Another aspect of the invention relates to computer program product in a computer usable medium for selecting an operating system at a target device 108, 110, 112 in communication with a server 104, 105. The product includes means for initiating 402 a network bootstrap program at the target device 108, 110, 112 and means for sending 406 a bootstrap list command from the target device 108, 110, 112 to the server 104, 105 (p. 9, lines 15-28). The product further includes means for receiving 516 an operating systems list of at least one operating system prior to executing an operating system at the target device 108, 110, 112, and means for selecting 518 a target operating system from the operating systems list. In the aspect of the invention, the target device 108, 110, 112 is to be remotely booted by the server 104, 105 (p. 1, lines 12-15).

**Claim 14**

In one embodiment, the product further includes means for relocating 524 the network bootstrap program after the target operating system is selected. (p. 11, lines 23-25)

**Claim 16**

In another embodiment, the target operating system is selected by a user (p.11, lines 13-18) of the target device 108, 110, 112.

**Claim 18**

A further aspect of the invention provides a network data processing system. The system includes means for initiating 402 a network bootstrap program at the target device 108, 110, 112 and means for sending 406 a bootstrap list command from the target device 108, 110, 112 to the server 104, 105 (p. 9, lines 15-28). The system further includes means for receiving 516 an operating systems list of at least one operating system prior to executing an operating system at the target device 108, 110, 112, and means for selecting 518 a target operating system from the operating systems list. In the aspect of the invention, the target device 108, 110, 112 is to be remotely booted by the server 104, 105 (p. 1, lines 12-15).

**Claim 21**

In one embodiment, the system further includes means for relocating 524 the network bootstrap program after the target operating system is selected. (p. 11, lines 23-25)

**Claim 23**

In another embodiment, the target operating system is selected by a user (p.11, lines 13-18) of the target device 108, 110, 112.

August 24, 2006

Case No.: AUS920000800US1 (9000/10)

Serial No.: 09/731,631

Filed: December 7, 2000

Page 10 of 22

6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-23 were rejected as unpatentable over Beelitz in view of Barr.

7. ARGUMENTS

**A. The Examiner rejected claims 1-23 as unpatentable under 35 U.S.C. §103(a) by Beelitz in view of Barr**

The §103(a) rejection of claims 1-23 has been traversed. In order to maintain this §103(a) rejection, each and every element of the claimed invention must be taught or suggested by the references alone or in combination. Because the references do not teach or suggest each and every element, this rejection must fall.

Claims 1, 11, and 18 require, *inter alia*, “initiating a network bootstrap program at the target device, sending a bootstrap list command from the target device to the server, receiving an operating systems list of at least one operating system prior to executing an operating system at the target device, and selecting a target operating system from the operating systems list, wherein the target device is to be remotely booted by the server.”

The Examiner correctly notes that Beelitz does not teach or suggest such an element, and instead cites to Barr for such teachings. Specifically, the Examiner asserts that it “would have been obvious to one of ordinary skill in the art at the time the invention was made to implement the Barr’s (sic) into the computer system of Beelitz to because (sic) it would have to (sic) provided an efficient system that lets a client and server to exchange (sic) a shared encryption key using the secure remote boot process.” See, ¶5 of the February 27, 2006 office action.

The Appellants traverse the Examiner’s assertion because Beelitz in view of Barr does not teach or suggest each claim limitation, and also because Beelitz specifically teaches away from such a combination, because there must be a reasonable expectation of success, and because the mere ability to combine references is insufficient to support a rejection.

**Beelitz in view of Barr does not teach or suggest each claimed element**

At most, Beelitz discloses that a targeted computer system 137 is initially booted up to perform the operations and instructions as per associated shell script files to load the selected programs onto its hard drive and to run the tests. Beelitz *does not teach*, as suggested by the Examiner, that the targeted computer system either “initiates a network

bootstrap program at the target device” or sends “a bootstrap list command from the target device to the server.” The Examiner cites to col. 14, line 66 to col. 15 line 7 for such a teaching, but such reliance is misplaced, as that citation merely teaches:

In one embodiment, the targeted computer system is connected through a network card to a server in manufacturing. The targeted computer system 137 is initially booted up to perform the operations and instructions as per associated shell script files to load the selected programs onto its hard drive and to run the tests. In one embodiment, the selected software programs and operating system can be 5 down loaded and installed on the targeted computer system via the Internet.

Thus, while Beelitz teaches that the target computer system is “booted up to perform the operations and instructions as per associated shell scripts to load the selected programs onto its hard drive,” Beelitz does not teach or suggest either “initiating a network bootstrap program at the target device” or “sending a bootstrap list command from the target device to the server,” as claimed in claim 1, 11 and 18. Barr does not cure this defect.

Similarly, Beelitz does not teach or suggest “receiving an operating systems list of at least one operating system prior to executing an operating system at the target device,” as further claimed in claims 1, 11, and 18. Strangely, the Examiner cites to the same section of Beelitz, but a review of the cited section does not reveal any such disclosure or teaching. Barr does not cure this defect.

Thus, the references, alone or in combination, do not teach each element of the claims.

**Beelitz teaches away from the combination**

Beelitz teaches a generation of a compatible order for a computer system. The Beelitz system is for specifying, ordering, and building a build-to-order computer system, such that a purchaser of a build-to-order computer system can buy and order such a computer over a computer network such as the Internet. See, abstract, Beelitz.

Beelitz specifically teaches “[I]n step 207, control 103 provides to the user interface 105 a list of the operating system types available.” (emphasis added) at column 7 55-56. Furthermore, “[I]n step 204, control 103 accesses the master data base 125 to

create or generate a list of operating system types available for the targeted computer system" (column 7 lines 35-37).

Control 103 "receives an indication *from a user interface* 105 indicating the desire to purchase a computer system." Column 7, lines 30-32 (emphasis added). Thus, *control 103 is distinguished from the targeted computer system 137* (FIG. 1 of Beelitz) *and the terminal or user interface 105* (FIG. 1 of Beelitz). One of ordinary skill in the art would recognize that sending a list of available operating systems to a user who is purchasing a build-to-order computer is dramatically different than sending a list of available operating systems to the build-to-order computer.

Beelitz teaches that it has been known to install software programs and to perform tests on computer systems *before they are shipped* to businesses or individual customers. The goal of software installation and testing, according to Beelitz, is to efficiently produce a useful, reliable, computer system which may be *delivered* to businesses and individuals free from errors and ready to run. Beelitz, column 1, lines 56-61.

One of ordinary skill in the art would recognize that shipping a computer to a business or individual customer requires *disconnecting* any network connections used for software installation and testing. Those of ordinary skill in the art would also recognize that disconnecting a network connection terminates the connection, and therefore disconnecting a network connection is inconsistent with a target device [is] to be remotely booted by the server. Thus, one of ordinary skill in the art could not possibly be motivated to make the modifications suggested by the Examiner, as Beelitz unequivocally teaches away from the combination.

This lack of motivation is further emphasized by the failure of Beelitz to denounce its generation of a compatible order for a computer system as less than an ideal method of generating a compatible order for a computer system, much less denounce its methods as less than ideal for booting a target computer system. In addition, Barr fails to denounce its methods for ensuring the integrity of remote boot client data as less than perfect, or teach the desirability of the modification suggested by the Examiner.

The Examiner's proffered motivation rings hollow. See, ¶4 of the Feb. 27, 2006 office action. Beelitz does not teach the desirability of exchanging shared encryption keys, and therefore one of ordinary skill in the art would not be motivated to consult any references teaching methods of sharing encryption keys based on the teachings of

Beelitz. In addition, Barr does not teach the desirability of generating a compatible order for a computer system, so one of ordinary skill in the art would not be motivated to consult references teaching any such methods. Use of such disparate references in this ad-hoc rejection would indicate the use of impermissible hindsight.

**There is no reasonable expectation of success**

Similarly, there is no reasonable expectation of success based on the combination of references. MPEP 2143.02, *In re Merck & Co.*, 800 F.2d 1091 (Fed. Cir. 1986). Here, modifying the Beelitz system to feature the remote boot taught by Barr would entirely defeat the purpose of Beelitz, and cannot lead to any reasonable expectation of success. Indeed, one of ordinary skill in the art would more likely find a reasonable expectation of *failure*.

The Examiner asserts that it “would have been obvious to one of ordinary skill in the art at the time the invention was made to implement the Barr’s (sic) into the computer system of Beelitz to because (sic) it would have to (sic) provided an efficient system that lets a client and server to exchange (sic) a shared encryption key using the secure remote boot process.” However, any such modification would in fact change the function or structure of the network, contrary to the Examiner’s assertions. The function and structure of the Beelitz network is to efficiently produce a useful, reliable, computer system which may be *delivered* to businesses and individuals free from errors and ready to run. Beelitz, column 1, lines 56-61. Modifying the function or structure of the Beelitz network would not result in the claimed invention, but would rather result in a business or individual purchasing a built-to-order computer that is not delivered to the purchasing business or individual, but rather a computer that remains at the factory, sitting on an assembly line. Such a result cannot be considered anything less than an abject failure from both the consumer side (the consumer does not receive the purchased computer) and the supplier side (inventory eternally accumulates until the customer dissatisfaction resulting from failure to receive the purchased computer destroys the business).

**The mere ability to combine references is insufficient to support a rejection**

The mere fact that Beelitz can be modified in view of Barr to obtain the claimed invention (which Appellants deny) does not render the resultant modification obvious unless the prior art also suggests the desirability of the combination. See, *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990) (Claims were directed to an apparatus for producing an aerated cementitious composition by drawing air into the cementitious composition by driving the output pump at a capacity greater than the feed rate. The prior art reference taught that the feed means can be run at a variable speed, however the court found that this does not require that the output pump be run at the claimed speed so that air is drawn into the mixing chamber and is entrained in the ingredients during operation. Although a prior art device "may be capable of being modified to run the way the apparatus is claimed, there must be a suggestion or motivation in the reference to do so." 916 F.2d at 682, 16 USPQ2d at 1432.). See also *In re Fritch*, 972 F.2d 1260, 23 USPQ2d 1780 (Fed. Cir. 1992) (flexible landscape edging device which is conformable to a ground surface of varying slope not suggested by combination of prior art references).

The basis for the Examiner's assertion is that "would have been obvious to one of ordinary skill in the art at the time the invention was made to implement the Barr's (sic) into the computer system of Beelitz to because (sic) it would have to (sic) provided an efficient system that lets a client and server to exchange (sic) a shared encryption key using the secure remote boot process." However, the Examiner simply cannot conclusively assert that an implementation of the remote boot feature taught by Barr would permit the ability to efficiently produce a useful, reliable, computer system which may be *delivered* to businesses and individuals free from errors and ready to run. See, Beelitz, column 1, lines 56-61. Indeed, implementing a remote boot feature, as allegedly taught by Barr, would simply not work in a Beelitz system and would destroy the principle of operation of Beelitz.

**Dependent claims 6, 8, 9, 14, 16, 21 and 23**

In addition to the above arguments, Beelitz in view of Barr fails to teach or suggest each and every element of claims 6, 8, 9, 14, 16, 21 and 23

Claims 6, 14, and 21 require “relocating the network bootstrap program after the target operating system is selected” as claimed in claims 6, 14, and 21. The Examiner’s citation to Beelitz, column 8, lines 40-45 is misplaced, as no such teaching is made. Instead, Beelitz discloses creating a system descriptor file that includes the specific part number for the selected version and language of the selected operating system, including a prefix. This disclosure simply cannot render claims for relocating the network bootstrap program after the target operating system is selected obvious. Barr does not cure this defect. Withdrawal of the rejections to claims 6, 14, and 21 is requested for at least this additional reason.

Claims 8, 16, and 23 require “the target operating system is selected by a user of the target device”. Beelitz in view of Barr does not teach such an element. At most, the Examiner cites to Beelitz, arguing that Beelitz teaches that the target operating system is determined by a “sniffing feature” to determine that the list of software programs is compatible with the hardware. Beelitz provides an example that the “software sniffer” determines that the target has a certain size RAM, and only presents operating systems that are compatible with the size of the RAM in the target. See, Beelitz, column 18 line 60 to column 19 line 3. Thus, Beelitz discloses that the target operating system is determined by a system configuration, while the instant application claims that the target operating system is selected by a user. Indeed, Beelitz does not teach that a user has any access whatsoever to the target device, as the “user” is purchasing a build-to-order computer, and does not yet have possession of the computer – thus, the Beelitz user is physically unable to select any operating systems at the target device. Barr does not cure this defect.

Claim 9 requires determining from a user profile, at least one available operating system.” Beelitz in view of Barr makes no such teaching. The Examiner relies solely on Beelitz, but Beelitz does not disclose a user profile at all. The Examiner’s reliance on column 7 lines 49-56 for such a disclosure is misplaced, as Beelitz merely discloses that a master database is consulted (see column 7 lines 36-41) at a server. No user profile is disclosed. Barr does not cure this defect.

Therefore, Appellants request the withdrawal of the rejections to claims 1, 11, and 18, as well as claims 2-10, 12-17, and 18-23 depending directly or indirectly from one of claims 1, 11, or 18.

**SUMMARY**

The Appellants respectfully submit that claims 1-23 fully satisfy the requirements of 35 U.S.C. §§102, 103 and 112. In view of the foregoing, favorable consideration and early passage to issue of the present application is respectfully requested.

Dated: August 24, 2006

Respectfully submitted,  
STEVEN M. FRENCH, et al.

/Frank C. Nicholas/

---

CARDINAL LAW GROUP  
1603 Orrington Avenue, Suite 2000  
Evanston, IL 60201  
(847) 905-7111

FRANK C. NICHOLAS,  
Registration No. (33,983)  
Attorney for Appellants

## 10. APPENDIX

1. A method of selecting an operating system at a target device in communication with a server, comprising:

initiating a network bootstrap program at the target device;  
sending a bootstrap list command from the target device to the server;  
receiving an operating systems list of at least one operating system prior to executing an operating system at the target device; and  
selecting a target operating system from the operating systems list, wherein the target device is to be remotely booted by the server.

2. The method of claim 1 further comprising:

receiving instructions for the target operating system.

3. The method of claim 1 further comprising:

requesting the instructions for the target operating system from the server.

4. The method of claim 1 further comprising:

booting the target operating system based on the instructions.

5. The method of claim 1 wherein the operating systems list includes a default operating system.

6. The method of claim 1 further comprising:

relocating the network bootstrap program after the target operating system is selected.

7. The method of claim 1 wherein the target operating system is determined from a configuration file of the target device.

8. The method of claim 1 wherein the target operating system is selected by a user of the target device.

9. The method of claim 8 further comprising:  
determining from a user profile, at least one -available operating system;  
and  
including the user-available operating system with the operating systems list.

10. The method of claim 1 further comprising:  
determining from a target device profile, at least one device-available operating system; and  
including the device-available operating system with the operating systems list.

11. Computer program product in a computer usable medium for selecting an operating system at a target device, comprising:  
means for initiating network bootstrap program code at the target device;  
means for receiving a command requesting an operating systems list of at least one operating system;  
means for sending the operating systems list to the target device before an operating system is executed at the target device; and  
means for receiving a selection of a target operating system from the operating systems list, wherein the target device is to be remotely booted by the server.

12. The program of claim 11 further comprising:  
means for sending the target operating system to the target device.
13. The program of claim 11 further comprising:  
means for determining a default operating system.
14. The program of claim 11 further comprising:  
means for relocating the network bootstrap program code after the target operating system is selected.
15. The program of claim 11 further comprising:  
means for determining the target operating system from a configuration file of the target device.
16. The program of claim 11 further comprising:  
means for receiving the selection of the target operating system from a user of the target device.
17. The program of claim 16 further comprising:  
means for determining at least one operating system available to the user.

18. A network data processing system comprising:
  - means for initiating a network bootstrap program at a target device, the target device to be remotely booted by the server;
  - means for sending a command requesting an operating systems list of at least one operating system;
  - means for receiving the operating systems list prior to executing an operating system at the target device; and
  - means for selecting a target operating system from the operating systems list at the target device.
19. The system of claim 18 further comprising:
  - means for receiving the target operating system at the target device.
20. The system of claim 18 further comprising:
  - means for executing the target operating system at the target device.
21. The system of claim 18 further comprising:
  - means for relocating the network bootstrap program after the target operating system is selected.
22. The system of claim 18 further comprising:
  - means for determining the target operating system from a configuration file of the target device.
23. The system of claim 18 further comprising:
  - means for determining the target operating system from input of a user of the target device.

**Evidence Appendix**

None

**Related Proceedings Appendix**

None.